

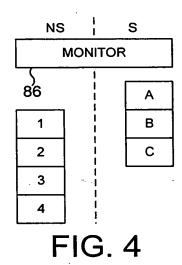
FIG. 2

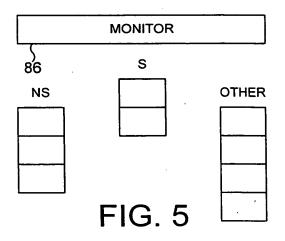
Inventor: WATT et al. SN 10/714,565/Sheet 2 of 56 Atty. Dkt.: 550-472

2 / 56

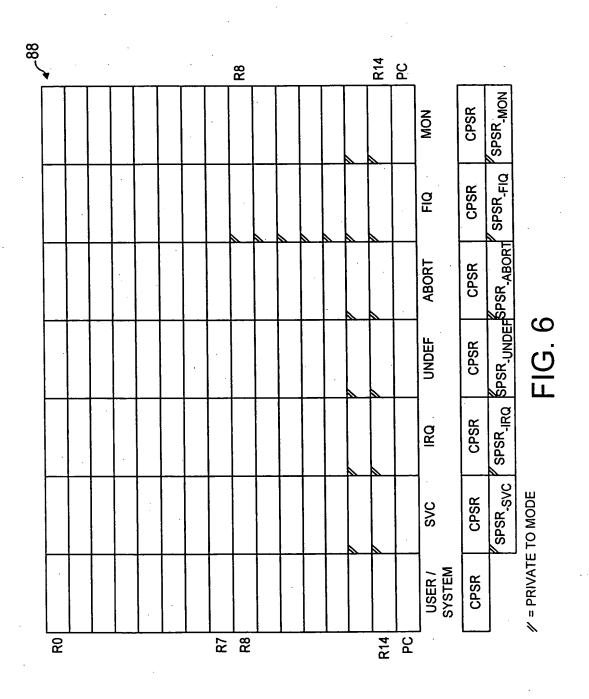
DOMAIN NON-SECURE SECURE MONITOR <u>MODE</u> NS MODE 1 S MODE 1 2 NS MODE 2 S MODE 2

FIG. 3

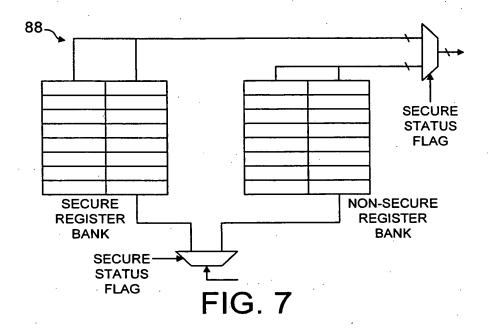


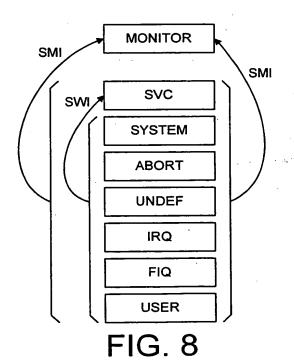


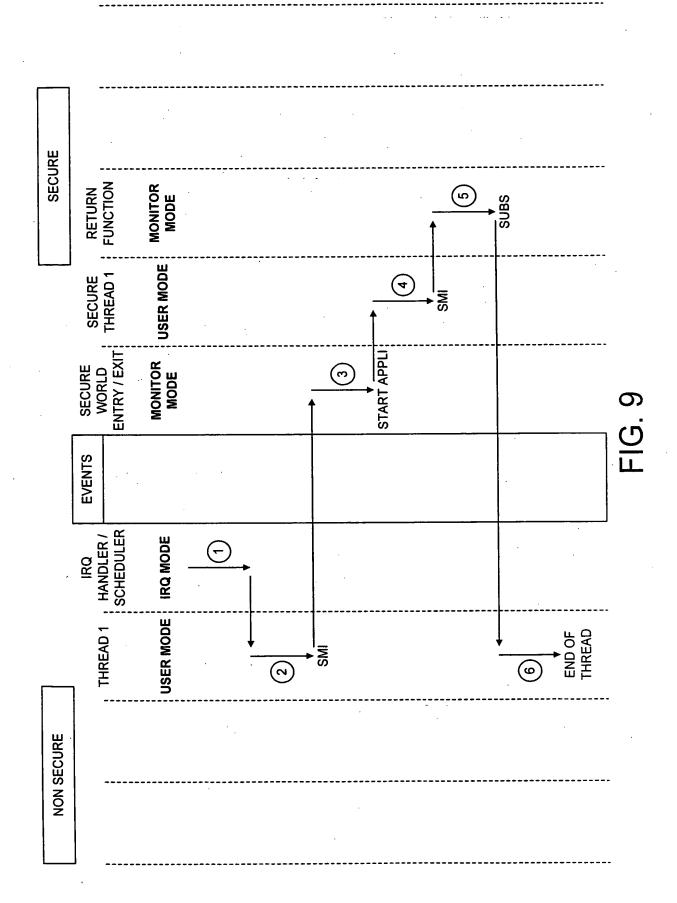
3 / 56



4 / 56







6 / 56

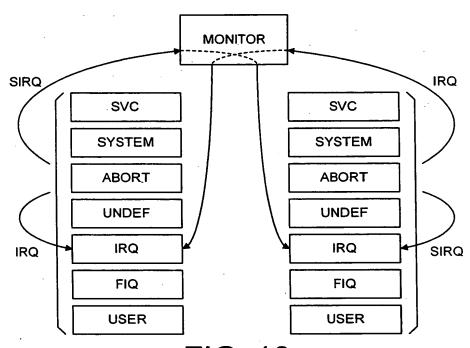
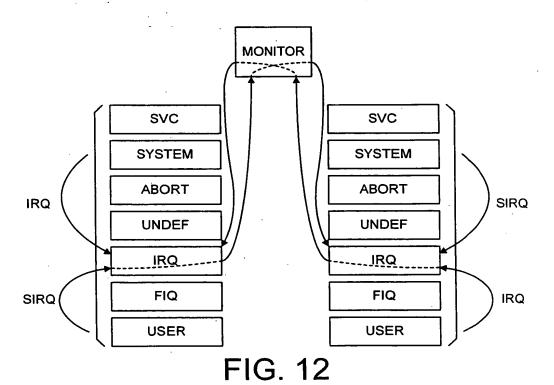
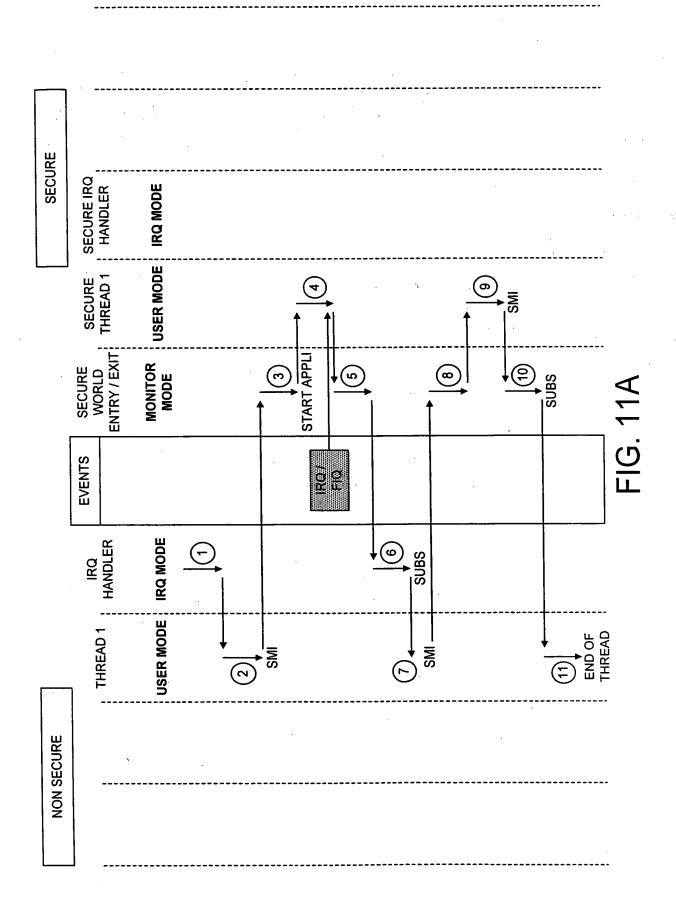
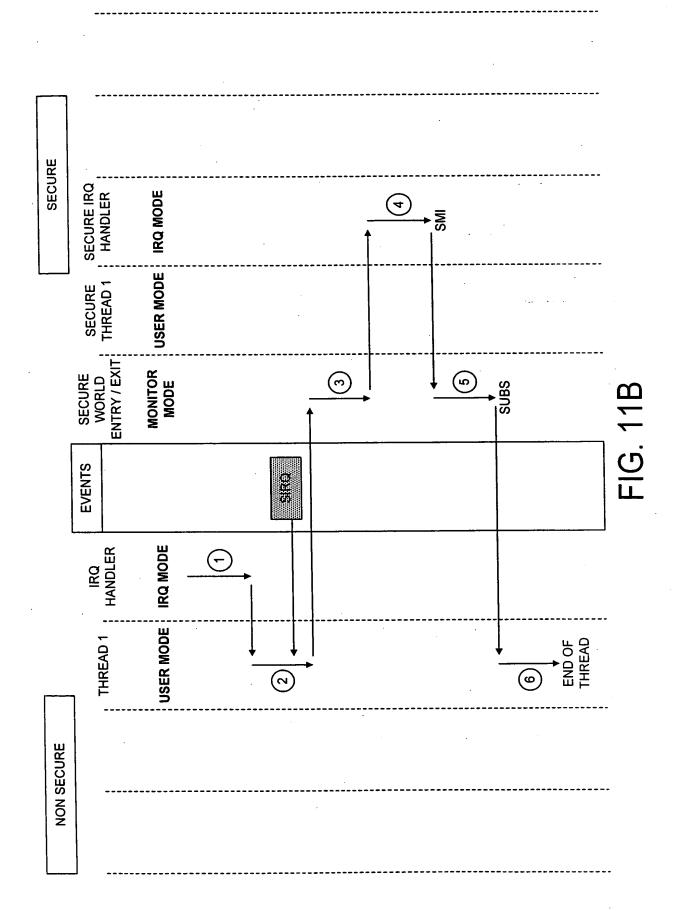


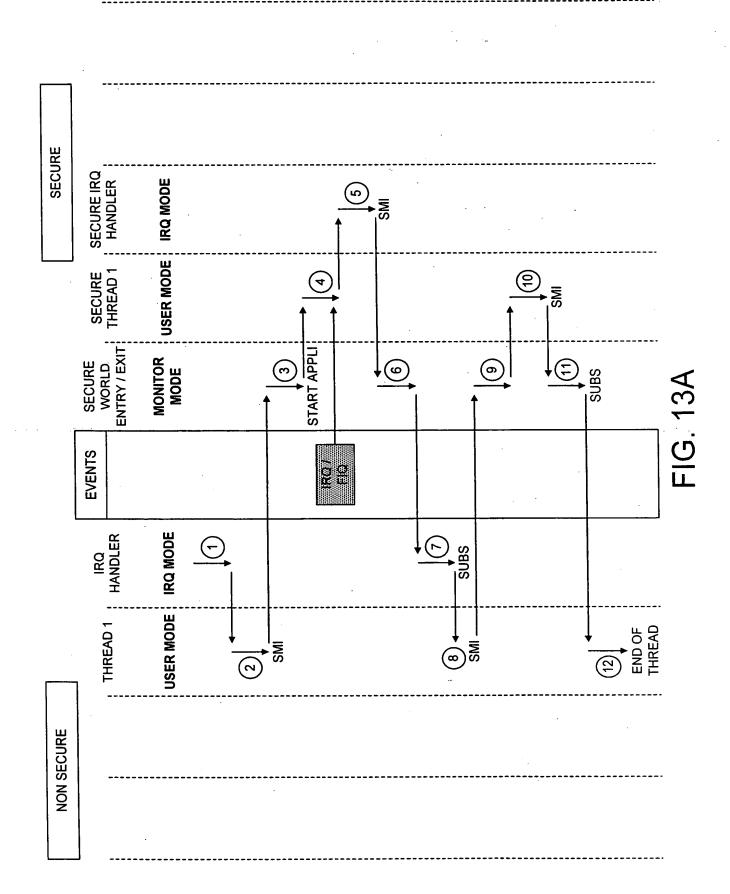
FIG. 10



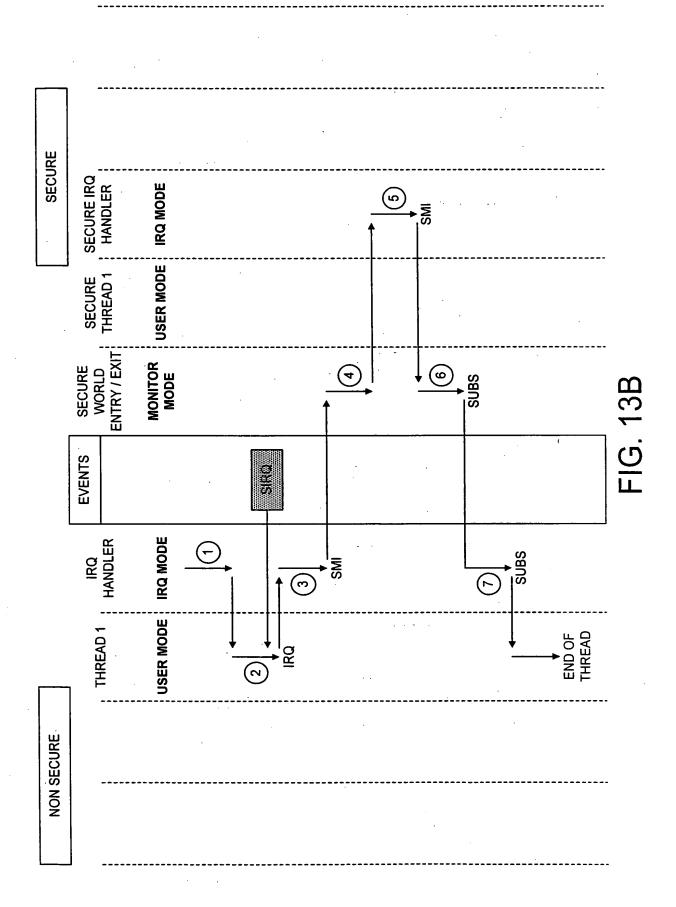
Inventor: WATT et al. SN 10/714,565/Sheet 7 of 56 Atty. Dkt.: 550-472







10 / 56



Inventor: WATT et al. SN 10/714,565/Sheet 11 of 56 Atty. Dkt.: 550-472

11 / 56

EXCEPTION	Valeire a Celarata	CORRESPONDING MODE
RESET	0x00	SUPERVISOR MODE
UNDEF	0x04	MONITOR MODE / UNDEF MODE
SWI	0x08	SUPERVISOR MODE / MONITOR MODE
PREFETCH ABORT	0x0C	ABORT MODE / MONITOR MODE
DATA ABORT	0x10	ABORT MODE / MONITOR MODE
IRQ / SIRQ	0x18	IRQ MODE / MONITOR MODE
FIQ	0x1C	FIQ MODE / MONITOR MODE
SMI	0x20	UNDEF MODE / MONITOR MODE

FIG. 14

MONITOR

RESET	VM0
UNDEF	VM1
SWI	VM2
PREFETCH ABORT	VM3
DATA ABORT	VM4
IRQ / SIRQ	VM5
FIQ	VM6
SMI	VM7

SECURE

RESET	VS0
UNDEF	VS1
SWI	VS2
PREFETCH ABORT	VS3
DATA ABORT	VS4
IRQ / SIRQ	VS5
FIQ	VS6
SMI	VS7

NON-SECURE

VNS0
VNS1
VNS2
VNS3
VNS4
VNS5
VNS6
VNS7

FIG. 15

12 / 56

CP15 MONITOR TRAP MASK REGISTER

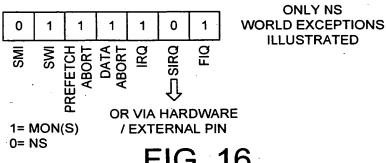
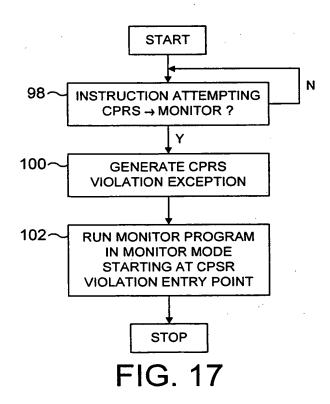
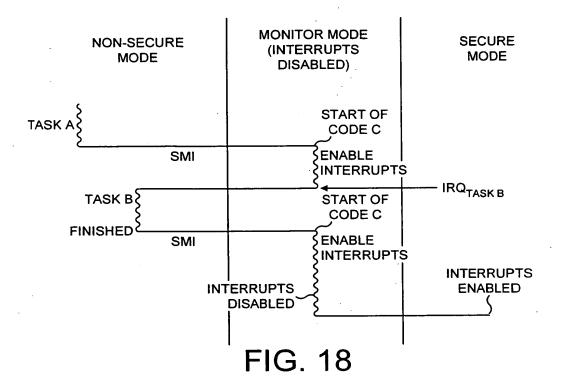
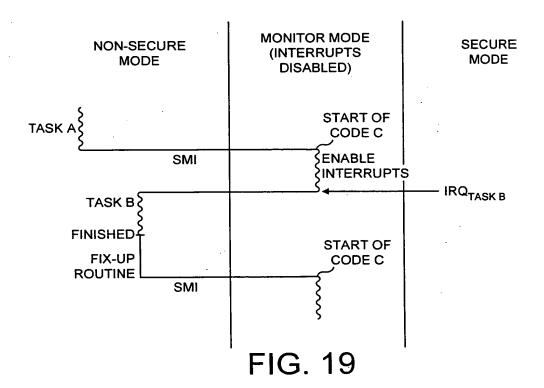


FIG. 16

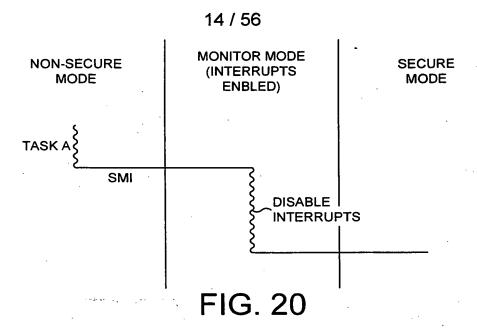


13 / 56





Inventor: WATT et al. SN 10/714,565/Sheet 14 of 56 Atty. Dkt.: 550-472



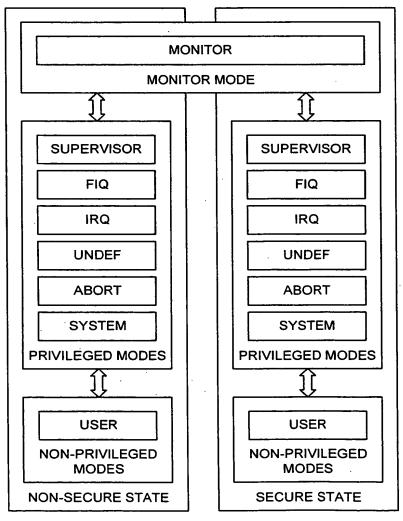
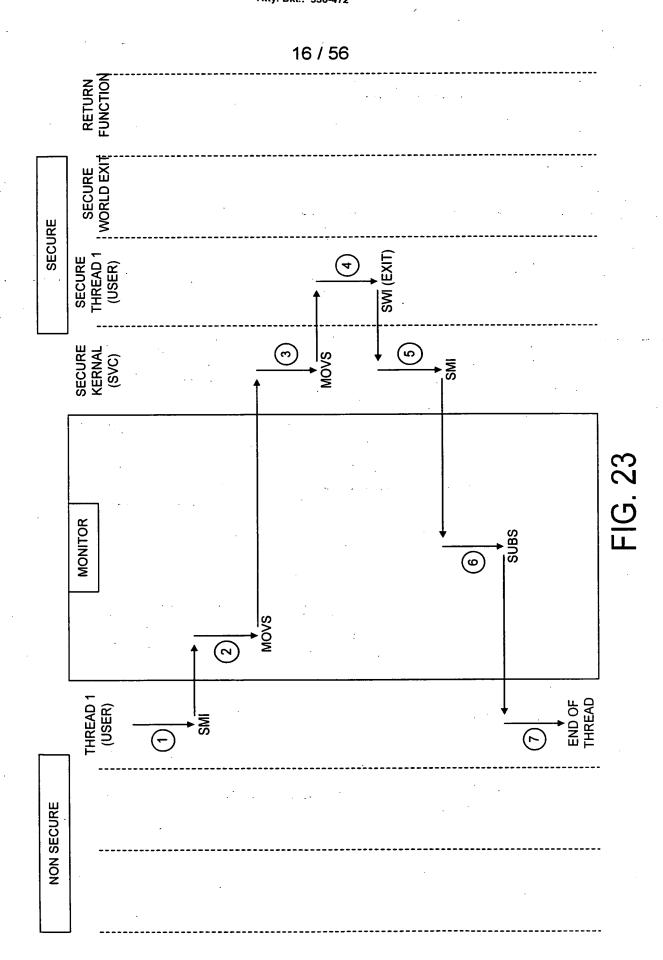


FIG. 21

Inventor: WATT et al. SN 10/714,565/Sheet 15 of 56 Atty. Dkt.: 550-472

																_	-		
MONITOR	R0	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13_MON	R14_MON	PC		CPSR	SPSR_MON
FAST INTERRUPT	R0	R1	R2	R3	R4	R5	R6	R7	R8_FIQ	R9_FIQ	R10_FIQ	R11_FIQ	R12_FIQ	R13_FIQ	R14_FIQ	PC		CPSR	SPSR_FIQ
INTERRUPT	R0	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13_IRQ	R14_IRQ	PC	÷	CPSR	SPSR_IRQ
UNDEFINED	RO	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13_UND	R14_UND	Эd		CPSR	SPSR_UND
ABORT	RO	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12		K:17 (2)?	PC		CPSR	SPSR_ABT
SUPERVISOR	RO	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12			PC		CPSR	SPSR_SVC
SYSTEM	RO	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	PC		CPSR	
USER	RO	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	PC:		CPSR	

FIG. 22



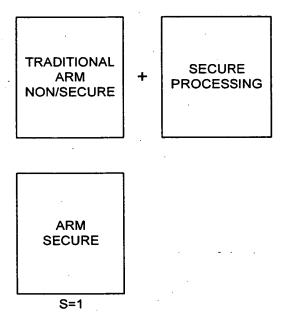
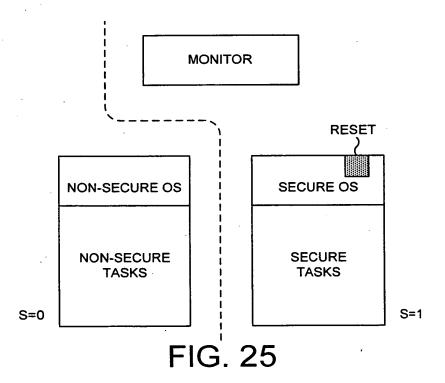
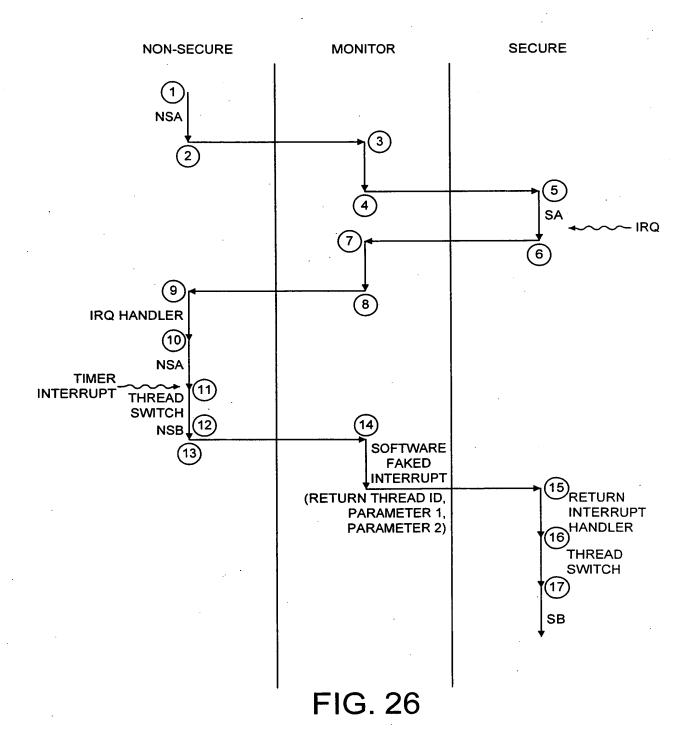


FIG. 24



18 / 56



19 / 56

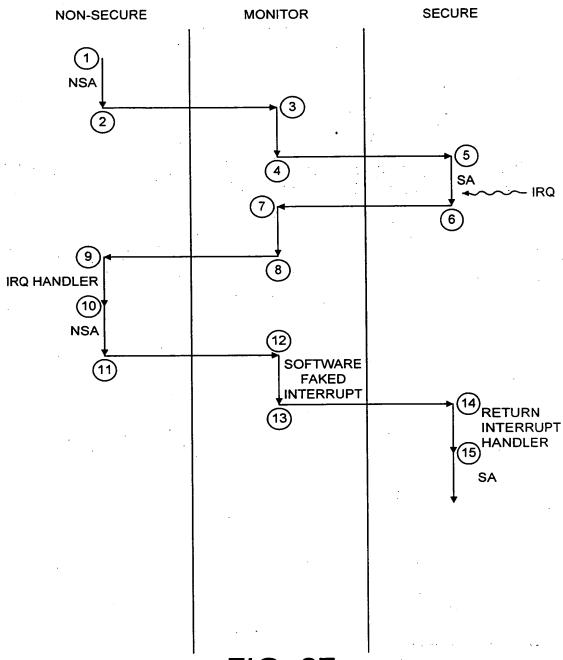


FIG. 27

Atty. Dkt.: 550-472

20 / 56

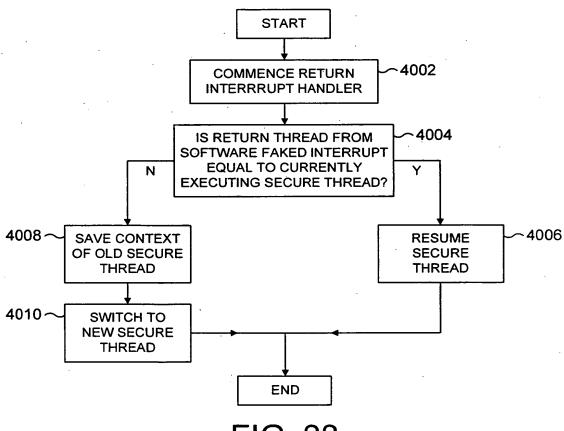


FIG. 28

21 / 56

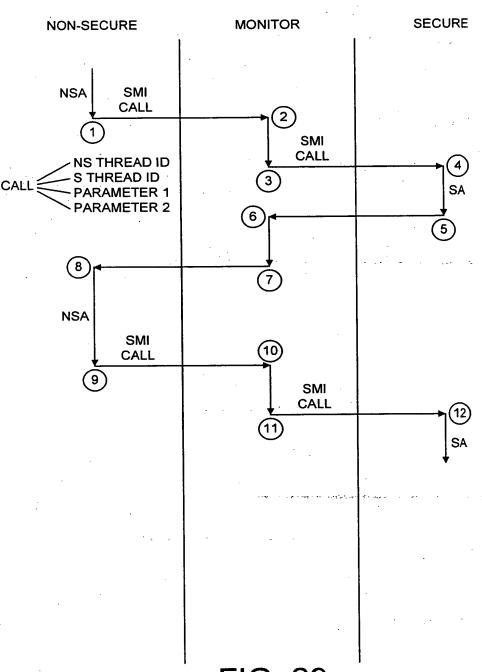


FIG. 29

22 / 56

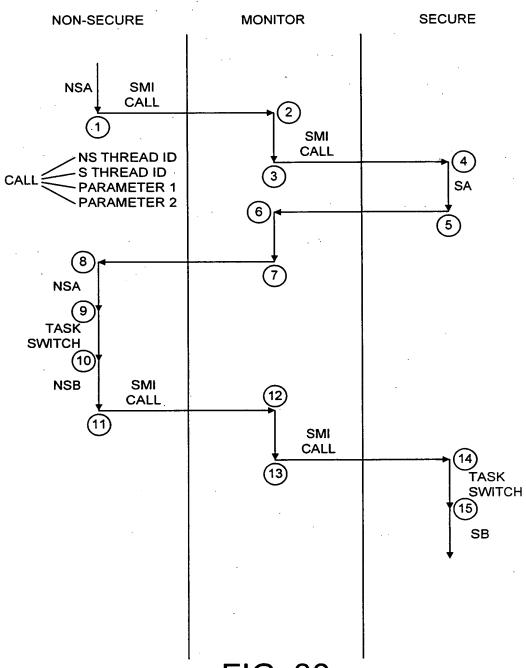


FIG. 30

23 / 56

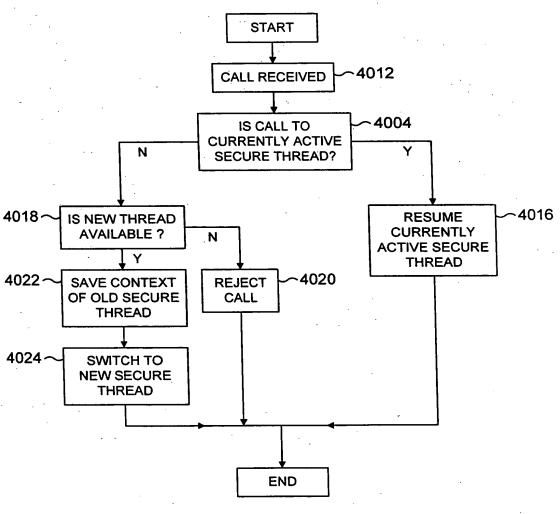


FIG. 31

24 / 56

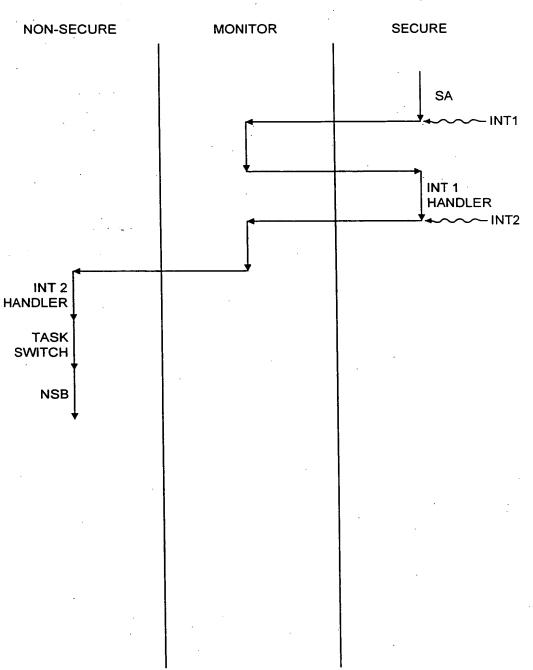
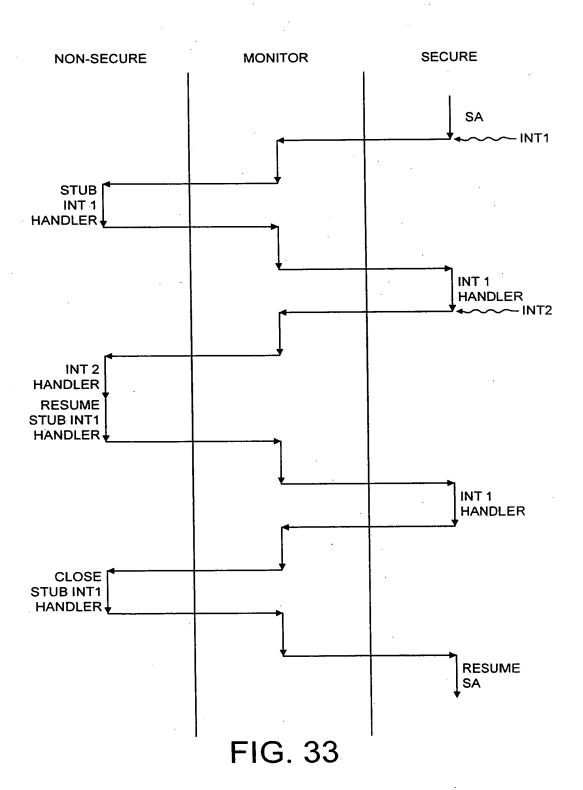


FIG. 32

25 / 56



26 / 56

INTERRUPT TYPE/PRIORITY	HOW HANDLED	_
1	S	
2	S	
3	NS	
4	NS/S	NO S ONLY
5	NS	HANDLERS LOWER THAN
6	NS/S	♦ HIGHEST NS
7 .	NS	HANDLER
•	•	
• .	•	
	_	

FIG. 34

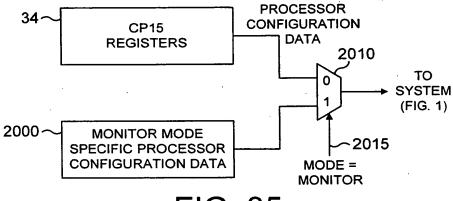


FIG. 35

27 / 56

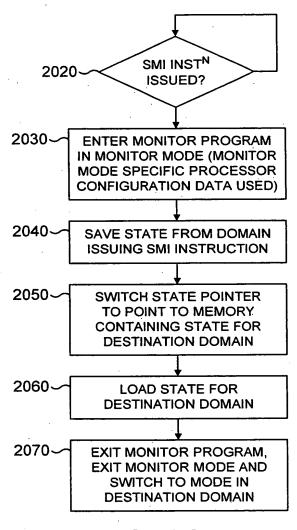
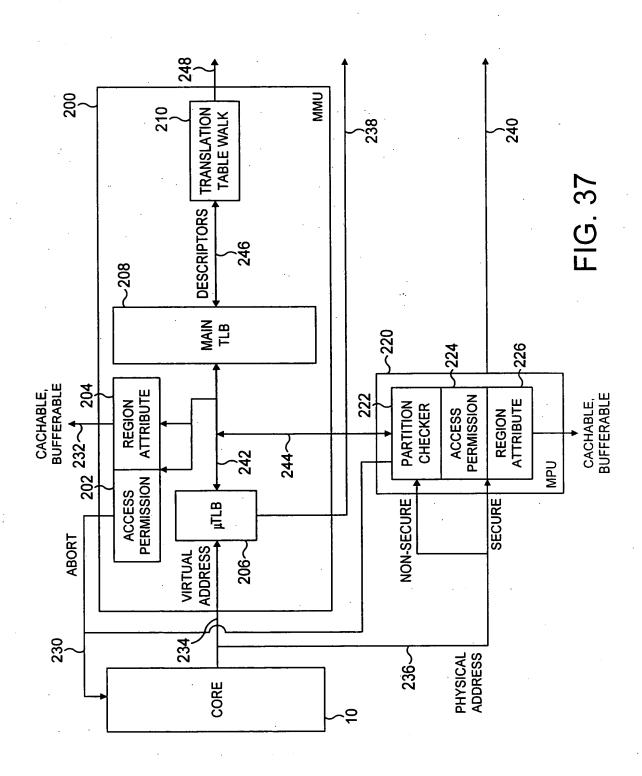
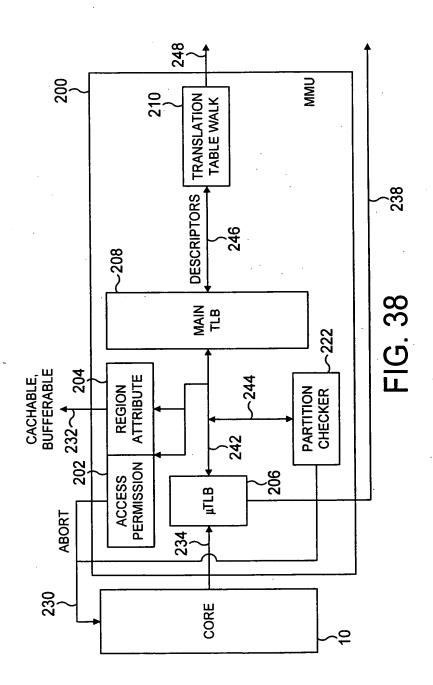


FIG. 36

28 / 56





30 / 56

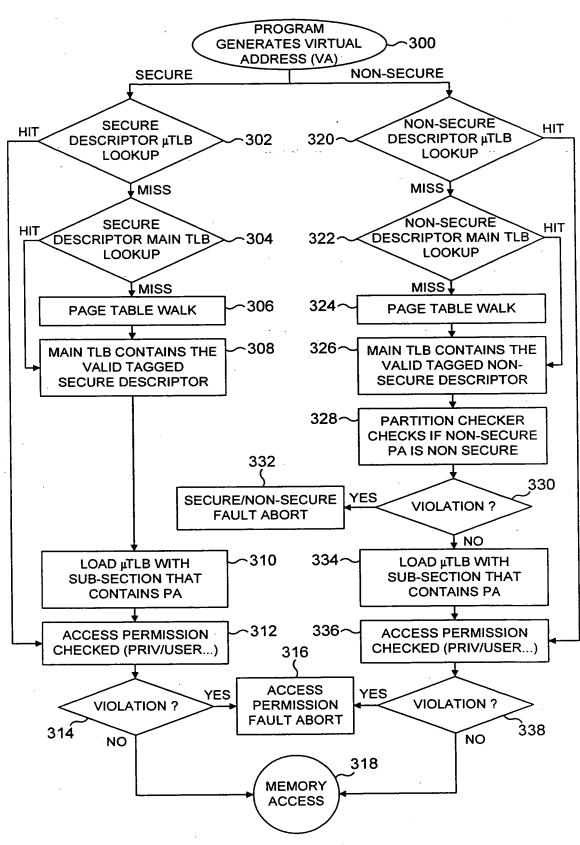
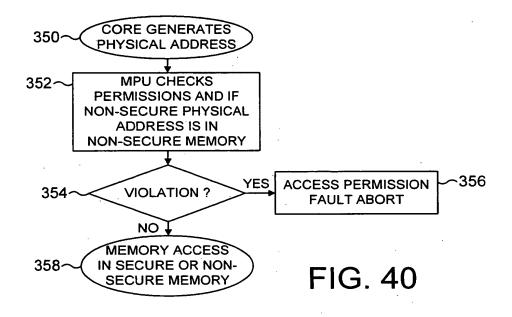
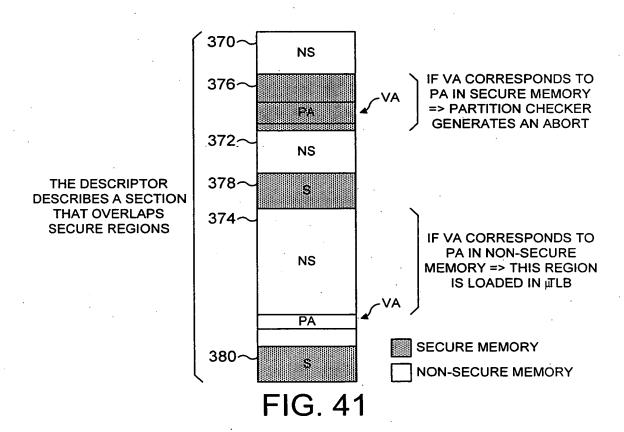


FIG. 39

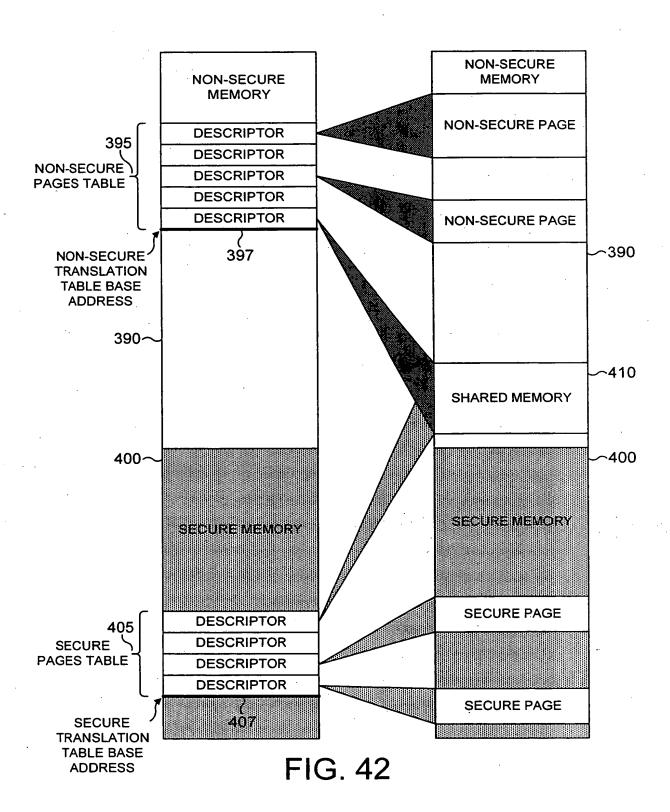
Inventor: WATT et al. SN 10/714,565/Sheet 31 of 56 Atty. Dkt.: 550-472

31 / 56





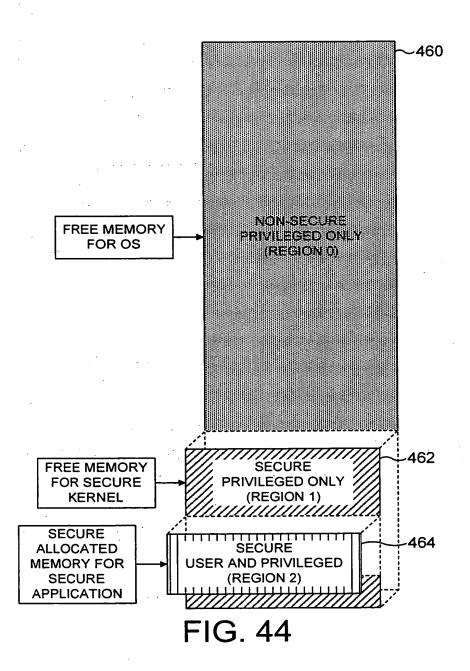
32 / 56



Inventor: WATT et al. SN 10/714,565/Sheet 33 of 56 Atty. Dkt.: 550-472

33 / 56 HIT OR MISS DESCRIPTOR ASID DOMAIN **BELONGING TO CURRENT** RUNNING PROCESS (ASID) (<u>)</u> ĊΩ 2 \$2 Ó ഗ U) DESCRIPTORS ī ă ď 63 **P**2 7 ā 7 ā 450~ **CURRENT ASID** ARE SELECTED DESCRIPTORS ASSOCIATED 2/ ONLY THE CHECK WITH THE DESCRIPTOR ASID DOMAIN S DESCRIPTORS
BELONGING TO CURRENT ഗ ഗ RUNNING WORLD ā E á 2 ā ī 7 7 445 $^{-}$ 440 \sim **440**~ **440** SELECTED 445~ 1/ AS THE $440 \sim$ SECURE 445~ DOMAIN, ONLY THE 445~ KERNEL ARE 440 DESCRIPTORS
BELONGING TO THE SECURE RUNNING IN CHECK **CORE IS** DESCRIPTOR ASID DOMAIN S SS SS SS SS g DESCRIPTORS CURRENTLY IN THE TLB ഗ ഗ S ഗ **P2 B**3 2 5 7 7 7 7 ᇟ TLB

34 / 56



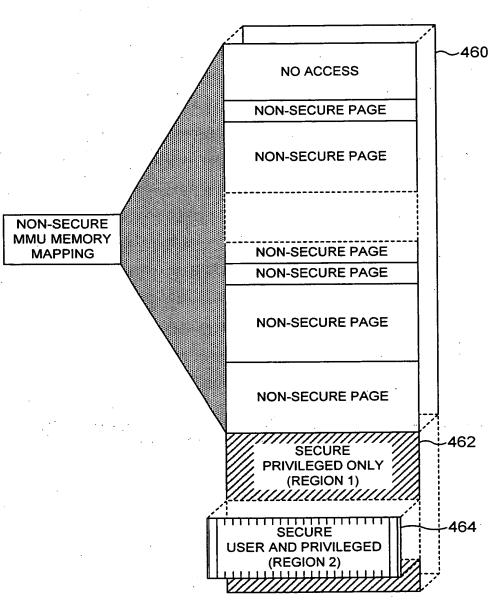


FIG. 45

Inventor: WATT et al. SN 10/714,565/Sheet 36 of 56 Atty. Dkt.: 550-472

36 / 56

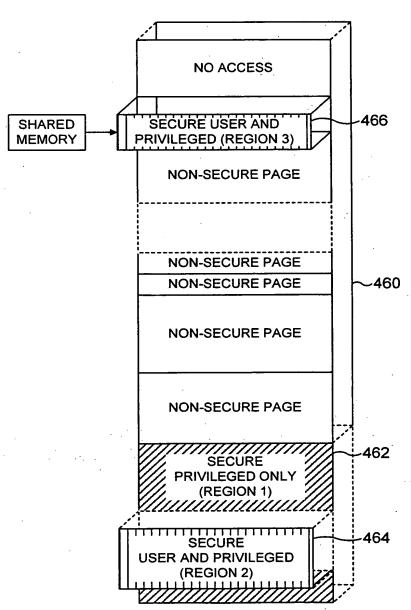


FIG. 46

37 / 56

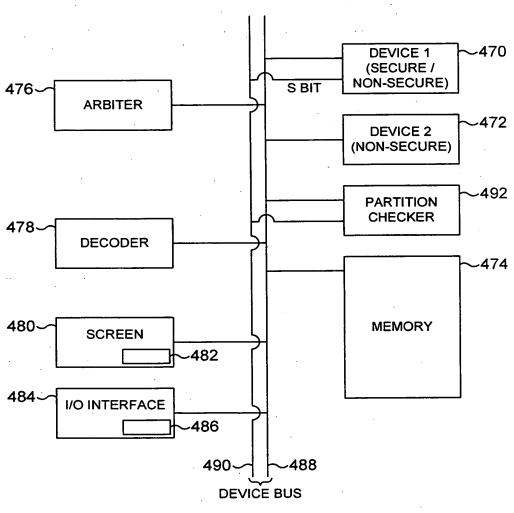


FIG. 47

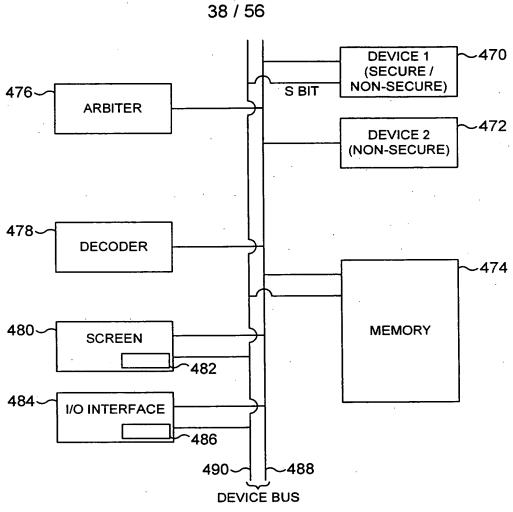


FIG. 48

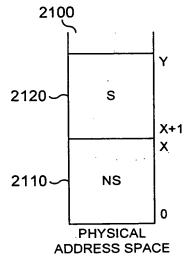
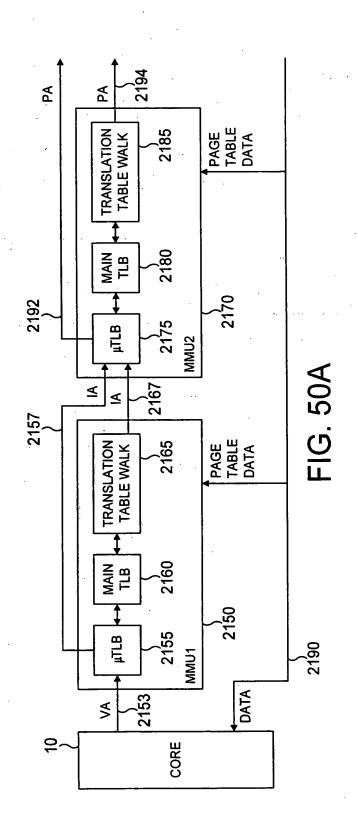
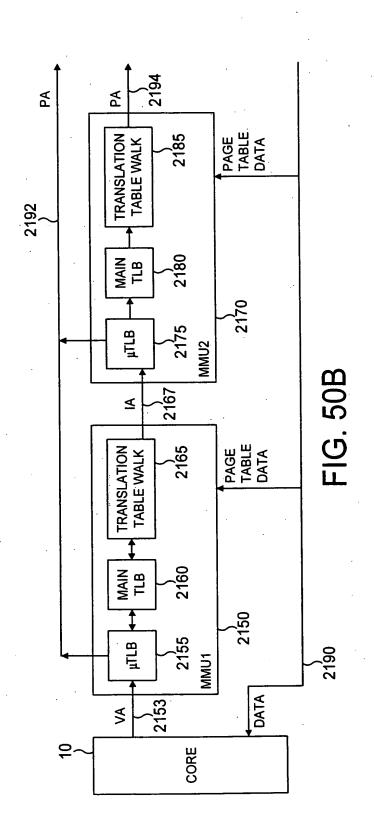


FIG. 49

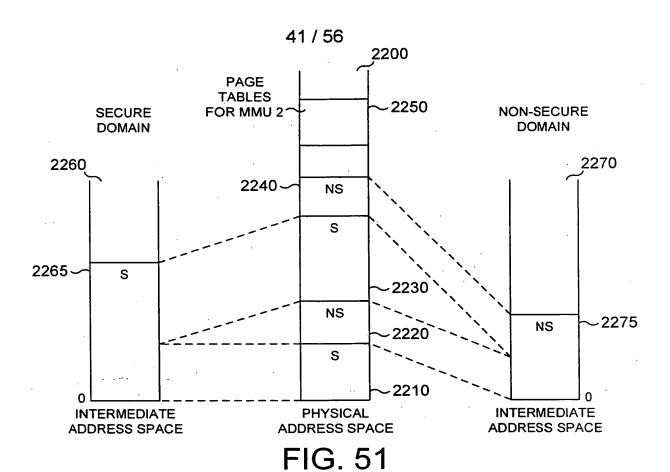
39 / 56

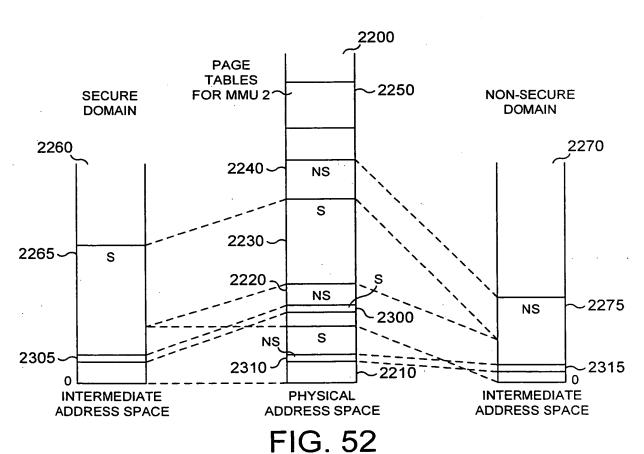


40 / 56



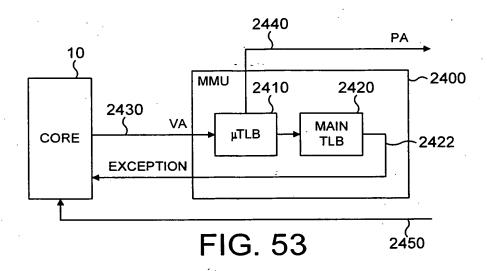
Inventor: WATT et al. SN 10/714,565/Sheet 41 of 56 Atty. Dkt.: 550-472





Inventor: WATT et al. SN 10/714,565/Sheet 42 of 56 Atty. Dkt.: 550-472

42 / 56



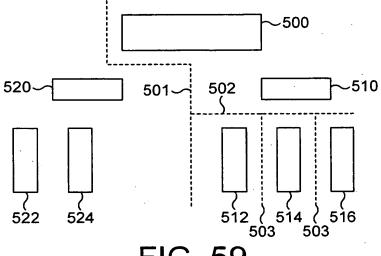


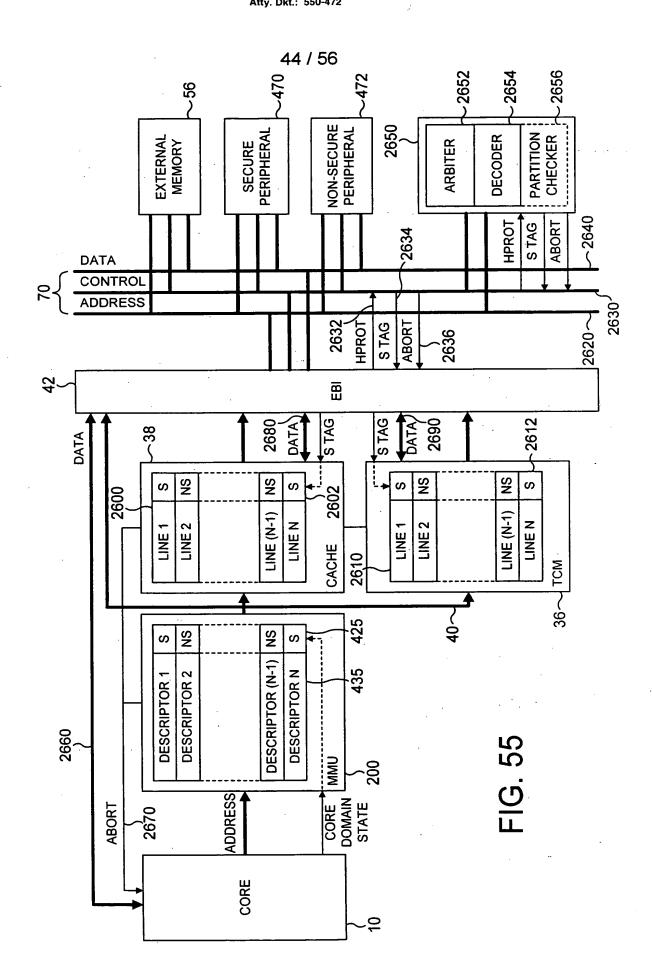
FIG. 59

Inventor: WATT et al. SN 10/714,565/Sheet 43 of 56

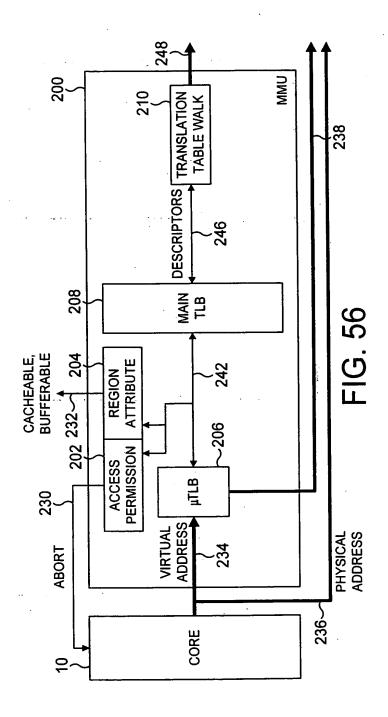
Atty. Dkt.: 550-472 43 / 56 TLB MISS Ν **EXCEPTION** 2500 DETECTED? 2510~ **ENTER MONITOR MODE AT** PREDETERMINED VECTOR FOR THE EXCEPTION 2520~ **RETRIEVE FAULTING VA** THAT CAUSED EXCEPTION 2530~ **DETERMINE IA FOR** RELEVANT FIRST **DESCRIPTOR IN FIRST TABLE** 2540~ REFERENCE SECOND TABLE TO OBTAIN PA FOR FIRST DESCRIPTOR 2550~ FETCH FIRST DESCRIPTOR TO **OBTAIN IA FOR FAULTING VA** 2560~ REFERENCE SECOND TABLE TO FIND SECOND **DESCRIPTOR GIVING PA FOR** IA OF FAULTING VA 2570~ **FETCH SECOND DESCRIPTOR TO OBTAIN PA** FOR FAULTING VA 2580 -MERGE FIRST AND SECOND **DESCRIPTORS TO GENERATE NEW DESCRIPTOR GIVING VA** TO PA TRANSLATION 2590~ STORE NEW DESCRIPTOR IN TLB 2595 RETURN FROM EXCEPTION

FIG. 54

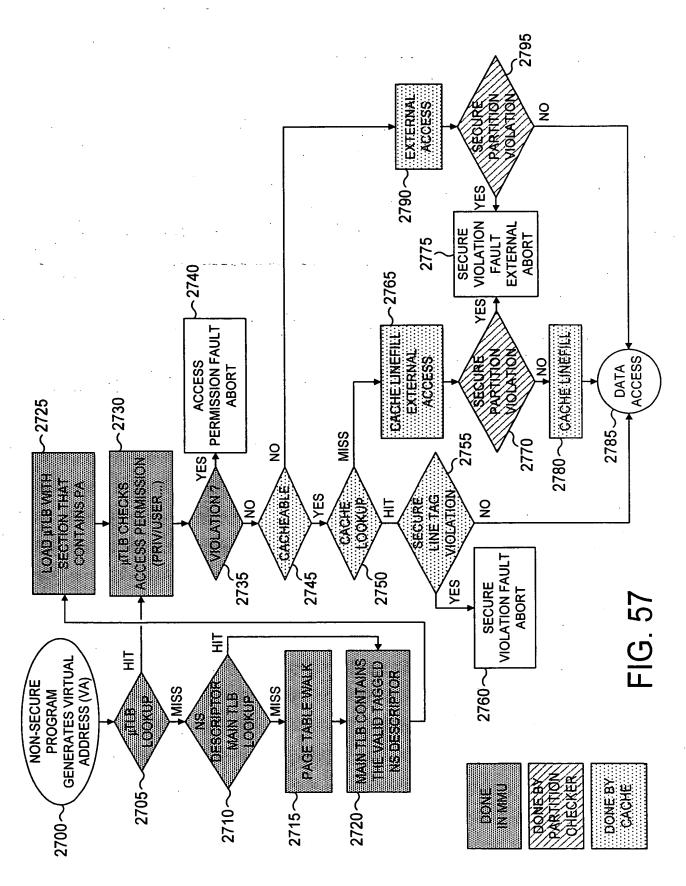
Inventor: WATT et al. SN 10/714,565/Sheet 44 of 56 Atty. Dkt.: 550-472



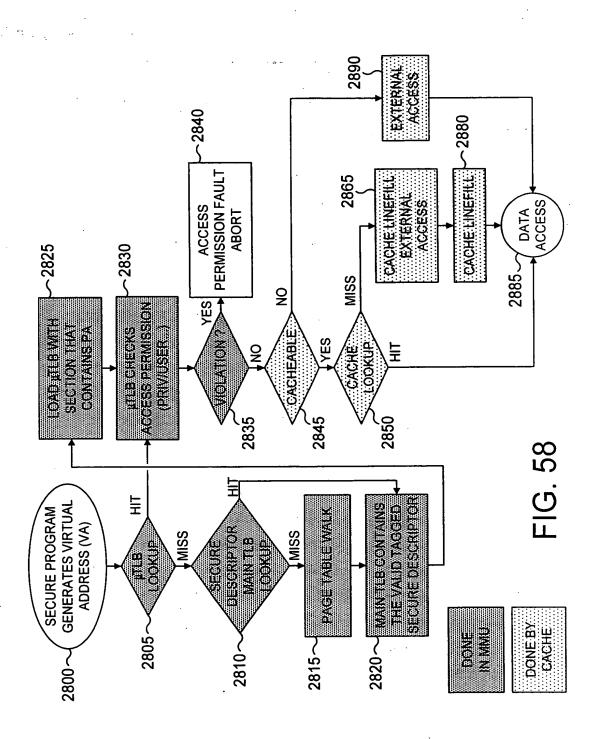
45 / 56



46 / 56



47 / 56



Inventor: WATT et al. SN 10/714,565/Sheet 48 of 56 Atty. Dkt.: 550-472

48 / 56

HALT	EDBGRQ INPUT PIN IS ASSERTED	NOT APPLICABLE	EXTERNAL DEBUG REQUEST
HALT	HALT INSTRUCTION HAS BEEN SCANNED IN	DEBUG TAP	
	INSTRUCTION ADDRESS AND/OR CP15 CONTEXT ID $\binom{2}{}$		-
HALT/MONITOR (1)	AND/OR CONTEXT-ID REGISTER AND COMPARISONS SUCCEED WITH	DEBUG TAP OR SOFTWARE (CP14)	os .
HALT/MONITOR	PROGRAM VECTOR TRAP REGISTER AND ADDRESS MATCHES	DEBUG TAP	
		(INSTRUCTION TRANSFER REGISTER) THROUGH DEBUG TAP OR USE BKPT INSTRUCTION DIRECTLY IN THE CODE	(INSTRUC REGIST DEBUG T INSTRUC
HALT/MONITOR	BKPT INSTRUCTION MUST REACH EXECUTION STAGE	PUT A BKPT INSTRUCTION INTO SCAN CHAIN 4	PUT A BK INTO S
	COMPARISONS SUCCEED WITH INSTRUCTION ADDRESS AND/OR CP15 CONTEXT ID (²)		
HALT/MONITOR (1)	PROGRAM BREAKPOINT REGISTER AND/OR CONTEXT-ID REGISTER AND	DEBUG TAP OR SOFTWARE (CP14)	DEB
ENTRY MODE	HOW TO ENTER?	HOW TO PROGRAM?	HOW

 $(^2)$: THE CORES HAVE SUPPORT FOR THREAD-AWARE BREAKPOINTS AND WATCHPOINTS IN ORDER TO ABLE TO ENABLE SECURE DEBUG ON SOME PARTICULAR THREADS. (1): IN MONITOR MODE, BREAKPOINTS AND WATCHPOINTS CANNOT BE DATA-DEPENDENT.

FIG. 60

49 / 56

NAME	MEANING	RESET VALUE	ACCESS	INSERTED IN SCAN CHAIN FOR TEST
MONITOR MODE ENABLE BIT	0: HALT MODE 1: MONITOR MODE	1	R/W BY PROGRAMMING THE ICE BY THE JTAG (SCAN 1) •R/W BY USING MRC/MCR	YES
			INSTRUCTION (CP14)	
SECURE DEBUG ENABLE BIT	0: DEBUG IN NON-SECURE WORLD ONLY 1: DEBUG IN	0	IN FUNCTIONAL MODE OR DEBUG MONITOR MODE:R/W BY USING MRC/ MCR INSTRUCTION (CP14) (ONLY IN SECURE SUPERVISOR MODE)	NO
	SECURE WORLD AND NON- SECURE WORLD		IN DEBUG HALT MODE: NO ACCESS - MCR/MRC INSTRUCTIONS HAVE ANY EFFECT	
			(R/W BY PROGRAMMING THE ICE BY THE JTAG (SCAN 1) IF JSDAEN=1	
SECURE TRACE ENABLE BIT	0: ETM IS ENABLED IN NON-SECURE WORLD ONLY. 1: ETM IS ENABLED IN SECURE WORLD AND NON- SECURE WORLD	0	IN FUNCTIONAL MODE OR DEBUG MONITOR MODE:R/W BY USING MRC/ MCR INSTRUCTION (CP14) (ONLY IN SECURE SUPERVISOR MODE) IN DEBUG HALT MODE: NO ACCESS- MCR/MRC INSTRUCTIONS HAVE ANY EFFECT (R/W BY PROGRAMMING THE ICE BY	NO
SECURE USER- MODE ENABLE BIT	0: DEBUG IS NOT POSSIBLE IN SECURE USER MODE 1: DEBUG IS POSSIBLE IN SECURE USER MODE	1	THE JTAG (SCAN 1) IF JSDAEN=1 IN FUNCTIONAL MODE OR DEBUG MONITOR MODE:R/W BY USING MRC/ MCR INSTRUCTION (CP14) (ONLY IN SECURE SUPERVISOR MODE) IN DEBUG HALT MODE: NO ACCESS- MCR/MRC INSTRUCTIONS HAVE ANY EFFECT (R/W BY PROGRAMMING THE ICE BY THE JTAG (SCAN 1) IF JSDAEN=1	NO
SECURE THREAD- AWARE ENABLE BIT	0: DEBUG IS NOT POSSIBLE FOR A PARTICULAR THREAD 1: DEBUG IS POSSIBLE FOR A PARTICULAR THREAD	0	IN FUNCTIONAL MODE OR DEBUG MONITOR MODE:R/W BY USING MRC/ MCR INSTRUCTION (CP14) (ONLY IN SECURE SUPERVISOR MODE) IN DEBUG HALT MODE: NO ACCESS - MCR/MRC INSTRUCTIONS HAVE ANY EFFECT (R/W BY PROGRAMMING THE ICE BY THE JTAG (SCAN 1) IF JSDAEN=1	NO

FIG. 61

50 / 56

FU	FUNCTION TABLE			
D	СК	Q[n+1]		
0		0		
1	\ \	1		
Х	/	Q[n]		

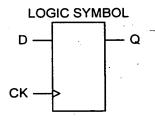


FIG. 62

FUNCTION TABLE					
D	S	SE	СК	Q[n+1]	
0	Х	0		0	
1	х	0		1	
X	Х	Х	/	Q[n]	
X	0	1		0	
X	1	1	/	1	

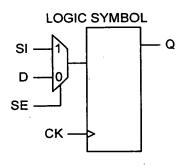
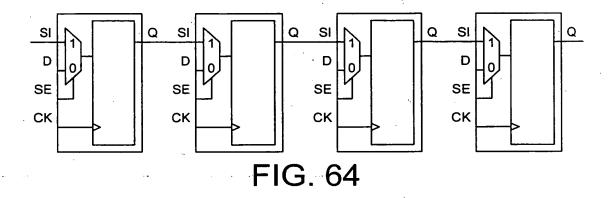
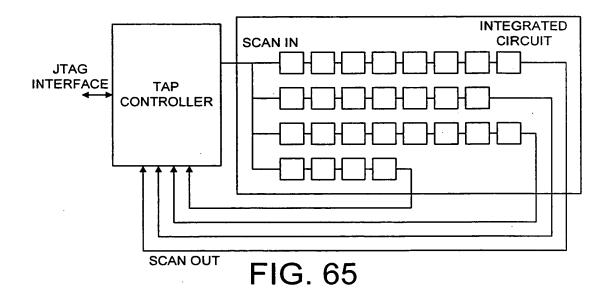


FIG. 63

Inventor: WATT et al. SN 10/714,565/Sheet 51 of 56 Atty. Dkt.: 550-472

51 / 56

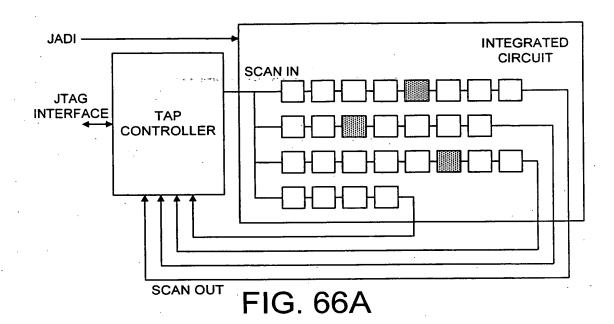


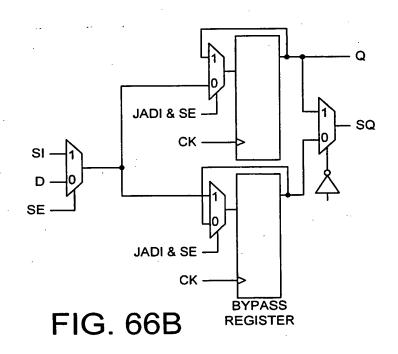


Inventor: WATT et al. SN 10/714,565/Sheet 52 of 56

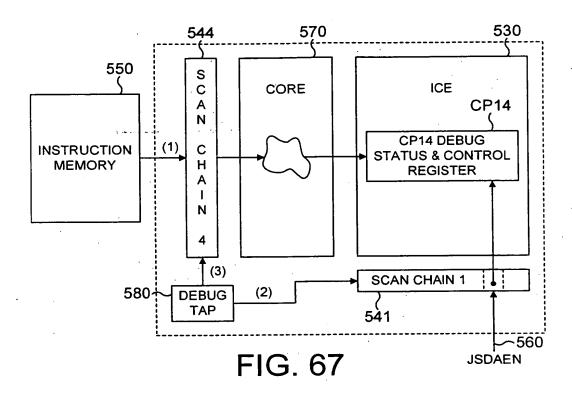
Atty. Dkt.: 550-472

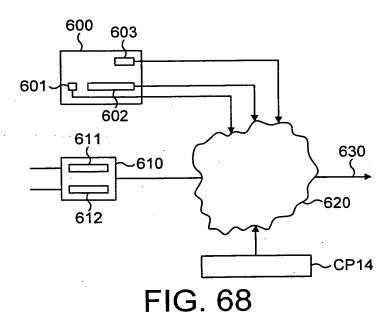
52 / 56





53 / 56





Inventor: WATT et al. SN 10/714,565/Sheet 54 of 56 Atty. Dkt.: 550-472

54 / 56

	MEANING	NO INTRUSIVE DEBUG IN ENTIRE WORLD IS POSSIBLE. ANY DEBUG REQUEST, BREAKPOINTS, WATCHPOINTS, AND OTHER MECHANISM TO ENTER DEBUG STATE ARE IGNORED IN ENTIRE SECURE WORLD	DEBUG IN ENTIRE SECURE WORLD IS POSSIBLE	DEBUG IN SECURE USER-MODE ONLY. ANY DEBUG REQUEST, BREAKPOINTS, WATCHPOINTS, AND OTHER MECHANISM TO ENTER DEBUG STATE ARE TAKEN INTO ACCOUNT IN USER MODE ONLY. (BREAKPOINTS AND WATCHPOINTS LINKED OR NOT TO A THREAD ID ARE TAKEN INTO ACCOUNT). ACCESS IN DEBUG IS RESTRICTED TO WHAT SECURE USER CAN HAVE ACCESS TO.	DEBUG IS POSSIBLE ONLY IN SOME PARTICULAR THREADS. IN THAT CASE ONLY THREAD-AWARE BREAKPOINTS AND WATCHPOINTS LINKED TO A THREAD ID ARE TAKEN INTO ACCOUNT TO ENTER DEBUG STATE. EACH THREAD CAN MOREOVER DEBUG ITS OWN CODE, AND ONLY ITS OWN CODE.
CP14 BITS IN DEBUG AND STATUS CONTROL REGISTER	SECURE THREAD- AWARE DEBUG ENABLE BIT	×	×	0	1
BUG AND STATUS	SECURE USER- MODE DEBUG ENABLE BIT	×	0	-	-
CP14 BITS IN DE	SECURE DEBUG ENABLE BIT	0		-	-

FIG. 69A

Inventor: WATT et al. SN 10/714,565/Sheet 55 of 56 Atty. Dkt.: 550-472

55 / 56

CP14 BITS IN DI	EBUG AND STATUS	CONTROL REGISTER	·
SECURE TRACE ENABLE BIT	SECURE USER- MODE DEBUG ENABLE BIT	SECURE THREAD- AWARE DEBUG ENABLE BIT	MEANING
0	Х	×	NO OBSERVABLE DEBUG IN ENTIRE SECURE WORLD IS POSSIBLE. TRACE MODULE (ETM) MUST NOT TRACE INTERNAL CORE ACTIVITY
1	0	×	TRACE IN ENTIRE SECURE WORLD IS POSSIBLE
1	1	0	TRACE IS POSSIBLE WHEN THE CORE IS IN SECURE USER-MODE ONLY
1	1	1	TRACE IS POSSIBLE ONLY WHEN THE CORE IS EXECUTING SOME PARTICULAR THREADS IN SECURE USER MODE. PARTICULAR HARDWARE MUST BE DEDICATED FOR THIS, OR RE-USE BREAKPOINT REGISTER PAIR: CONTEXT ID MATCH MUST ENABLE TRACE INSTEAD OF ENTERING DEBUG STATE

FIG. 69B

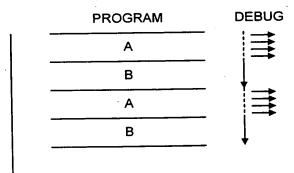


FIG. 70

Inventor: WATT et al. SN 10/714,565/Sheet 56 of 56

Atty. Dkt.: 550-472

56 / 56

METHOD OF ENTRY	ENTRY WHEN IN NON-SECURE WORLD	ENTRY WHEN IN SECURE WORLD
BREAKPOINT HITS	NON-SECURE PREFETCH ABORT HANDLER	SECURE PREFETCH ABORT HANDLER
SOFTWARE BREAKPOINT	NON-SECURE PREFETCH ABORT HANDLER	SECURE PREFETCH ABORT HANDLER
VECTOR TRAP BREAKPOINT	DISABLED FOR NON-SECURE DATA ABORT AND NON-SECURE PREFETCH ABORT INTERRUPTIONS. FOR OTHER NON-SECURE EXCEPTIONS, PREFETCH ABORT	DISABLED FOR SECURE DATA ABORT AND SECURE PREFETCH ABORT EXCEPTIONS (1) FOR OTHER EXCEPTIONS, SECURE PREFETCH ABORT
WATCHPOINT HITS	NON-SECURE DATA ABORT HANDLER	SECURE DATA ABORT HANDLER
INTERNAL DEBUG REQUEST	DEBUG STATE IN HALT MODE	DEBUG STATE IN HALT MODE
EXTERNAL DEBUG REQUEST	DEBUG STATE IN HALT MODE	DEBUG STATE IN HALT MODE

⁽¹⁾ SEE INFORMATION ON VECTOR TRAP REGISTER

FIG. 71A

METHOD OF ENTRY	ENTRY IN NON-SECURE WORLD	ENTRY IN SECURE WORLD
BREAKPOINT HITS	NON-SECURE PREFETCH ABORT HANDLER	BREAKPOINT IGNORED
SOFTWARE BREAKPOINT INSTRUCTION	NON-SECURE PREFETCH ABORT HANDLER	INSTRUCTION IGNORED (1)
VECTOR TRAP BREAKPOINT	DISABLED FOR NON-SECURE DATA ABORT AND NON-SECURE PREFETCH ABORT INTERRUPTIONS. FOR OTHER INTERRUPTION NON-SECURE PREFETCH ABORT	BREAKPOINT IGNORED
WATCHPOINT HITS	NON-SECURE DATA ABORT HANDLER	WATCHPOINT IGNORED
INTERNAL DEBUG REQUEST	DEBUG STATE IN HALT MODE	REQUESTIGNORED
EXTERNAL DEBUG REQUEST	DEBUG STATE IN HALT MODE	REQUEST IGNORED
DEBUG RE-ENTRY FROM SYSTEM SPEED ACCESS	NOT APPLICABLE	NOT APPLICABLE

 $^(^1)$ AS SUBSTITUTION OF BKPT INSTRUCTION IN SECURE WORLD FROM NON-SECURE WORLD IS NOT POSSIBLE, NON-SECURE ABORT MUST HANDLE THE VIOLATION.

 $^(^2)$ NOTE THAT WHEN EXTERNAL OR INTERNAL DEBUG REQUEST IS ASSERTED, THE CORE ENTERS HALT MODE AND NOT MONITOR MODE